REMARKS

Claims 1-29 are pending in the present application. By this Response, claims 1, 11 and 20 are amended for clarification to recite loading a plurality of predicate registers with "values associated with a plurality of branch addresses" and calling an instruction associated with one of the "plurality of branch addresses based on the values of the plurality of predicate registers." No new matter has been added. Support for the amendments to the claims may be found at least on pages 13-14 of the present specification. Reconsideration of the claims is respectfully requested.

I. Allowable Subject Matter

Applicant thanks Examiner Harkness for the indication of allowable subject matter in claims 2-6, 10, 12-16, 19, 21-25 and 29. However, for the reasons set forth herein below, Applicant respectfully submits that all of the claims are directed to allowable subject matter and that the application is in condition for allowance.

II. Objection to the Drawings

The Office Action objects to the drawings stating that the limitations in the claims dealing with a range of branch addresses, using registers to set the predicate registers, and a mask must be shown in the figures. Applicant respectfully submits that these features are shown in Figure 3. Looking at Figure 3, when read in conjunction with the specification, the features of a range of branch addresses and using registers to set predicate registers is clearly shown in steps 330-360. Steps 330 and 340 clearly show the setting of register values to indicate a range of addresses in a target address array (register A) and an index (register B). Step 350 clearly shows shifting of the register A value by the register B value, i.e. indexing into the range of addresses. Step 360 then shows moving a range of bits from the value of register A into the predicate registers. Thus, the plurality of addresses are illustrated as the value for register A and the setting of predicate registers based on other registers, e.g., regA and regB, is clearly shown.

The specification states on page 13, lines 22-29 that the movement of the bits in register A is performed using a mask that identifies a contiguous range of predicate registers. Thus, the use of the mask is illustrated in step 360 of Figure 3. However, in order to explicitly mention a mask in the figure, step 360 is amended by the attached replacement sheet for Figure 3 to indicate that bits in register A are moved to the predicate registers based on a mask. Thus, the figures clearly illustrate all of the claimed features that the Office Action alleges are not illustrated. Accordingly, Applicant respectfully requests withdrawal of the objection to the figures.

III. Claim Objections

The Office Action alleges that claims 2, 12 and 21 are unclear. Specifically, the Office Action alleges that it is unclear that the terminology "2**(lowpredicate-low value)" means 2 to the power of (lowpredicate-low value). Applicant respectfully submits that the use of the "**" symbol to indicate "to the power of" is a widely accepted convention and is not unclear. Moreover, it is apparent from the Office Action that the Examiner understood that this terminology meant 2 to the power of (lowpredicate-low value) and thus, one of ordinary skill in the art would understand this terminology in the same manner as the Examiner. Accordingly, Applicant respectfully submits that this terminology is not unclear and respectfully requests withdrawal of the objection to the claims. If however, the Examiner has some suggestion as to how the claims should be amended to allegedly make this terminology clearer, Applicant respectfully requests that the Examiner provide his suggestion in the next communication.

IV. 35 U.S.C. § 112, First Paragraph

The Office Action rejects claims 1-29 under 35 U.S.C. § 112, first paragraph, as failing to comply with the enablement requirement. This rejection is respectfully traversed.

As to claims 1-29, the Office Action states:

5. Claims 1-29 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to enable one skilled in the art to which it pertains, or with which it is most nearly connected, to make/or use the invention. In independent claims 1, 11, and 29, Applicant has claimed "loading a plurality of predicate registers with branch addresses" while the invention according to the specification values from regA is moved to the predicate register set, where predicate registers are only one bit registers, indicating whether the instruction should be executed or not (Spec. page 13 lines 8-24). There is no description of any branch addresses being loaded into any registers found in the specification. It is assumed that the claim is being limited by loading a single branch address, which is greater than 1 bit, into a single one bit predicate register.

Office Action dated September 25, 2003, page 3.

Claim 1, which is representative of the other rejected independent claims 11 and 20, is amended to read as follows:

1. A method of implementing a switch instruction in an IA64 architecture based data processing device, comprising:

receiving a call to the switch instruction, the call including one or more parameters for the switch instruction;

loading a plurality of predicate registers with values associated with a plurality of branch addresses based on the one or more parameters; and

calling an instruction associated with one of the plurality of branch addresses based on the values of the plurality of predicate registers.

As is clear from pages 13 and 14 of the present specification, the present invention uses a range of predicate registers that is the same size as the number of entries in a target branch address array. If the value of a predicate register is 1, a branch identified by that predicate register is executed (page 14, lines 9-12). Thus, when the value of the shifted register A, i.e. the bit values of register A, is moved into the predicate registers, various ones of the predicate registers will be set to 1 and the others will have their values set to 0 based on the value of the shifted register A. Thus, by moving the shifted register A value into the predicate registers, values associated with various ones of the branch addresses are loaded into the predicate registers. Which predicate registers are

loaded with values is determined based on the parameters that are received, e.g., in one exemplary embodiment parameters that identify a default target address, a low value, a high value, and an array of target addresses (page 12, lines 1-5).

Thus, it is clear from the above, and the clarifying amendments to the claims, that there is ample support in the specification for the features recited in the independent claims. Furthermore, the Office Action's assumption that only one branch address is being loaded into the predicate registers is incorrect and the present claims should not be limited to the Examiner's assumption. Accordingly, Applicant respectfully requests withdrawal of the rejection of claims 1-29 under 35 U.S.C. § 112, first paragraph.

V. 35 U.S.C. § 102, Alleged Anticipation of Claims 1, 9, 11, 18, 20 and 28

The Office Action rejects claims 1, 9, 11, 18, 20, and 28 under 35 U.S.C. § 102 as being allegedly anticipated by Maslennikov et al. (U.S. Patent Number 6,412,105). This rejection is respectfully traversed.

As to independent claims 1, 11, and 20, the Office Action states:

> Referring to claims 1, 11, and 20 Maslennikov has taught a method of implementing a switch instruction in an IA64 architecture based data processing device, comprising:
>
> Receiving a call to the switch instruction, the call including one or more parameters for the switch instruction (Maslennikov figure 1column 3 lines 21-40; inherently the switch instruction would have to be called);
>
> Loading a plurality of predicate registers with branch addresses based on the one or more parameters (Maslennikov figure 1 column 3 lines 32-40; the predicate registers would inherently be loaded with the parameters that the compiler assigned to show which branch of the switch would be the correct path); and
>
> Calling an instruction associated with one of the branch addresses based on values of the plurality of predicate registers (Maslennikov figure 1 column 3 lines 32-40; which ever branch instruction of the switch statement has its predicate value set to true will be the branch that is called).

Office Action dated September 25, 2003, page 4.

Claim 1, which is representative of the other rejected independent claims 11 and 20 with regard to similarly recited subject matter, reads as follows:

> 1. A method of implementing a switch instruction in an IA64 architecture based data processing device, comprising:
> receiving a call to the switch instruction, the call including one or more parameters for the switch instruction;
> loading a plurality of predicate registers with values associated with a plurality of branch addresses based on the one or more parameters; and
> calling an instruction associated with one of the plurality of branch addresses based on the values of the plurality of predicate registers. (emphasis added)

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994). Anticipation focuses on whether a claim reads on the product or process a prior art reference discloses, not on what the reference broadly teaches. *Kalman v. Kimberly-Clark Corp.*, 713 F.2d 760, 218 U.S.P.Q. 781 (Fed. Cir. 1983). Applicant respectfully submits that Maslennikov does not identically show every element of the claimed invention arranged as they are in the claims.

Maslennikov teaches a method and apparatus for compilation of multi-way decisions. In the system of Maslennikov, profile data is compiled and the most probable alternatives for a switch instruction in code are identified. Based on the identified most probable alternatives for a switch statement in code, the switch statement is transformed such that the most probable alternatives are moved out of the transformed switch statement and appear before the switch statement in an if-else condition (see column 4, lines 1-63). While Maslennikov states that the alternatives that are subsequently moved out of the switch statement may be transformed to the "predicate form" (column 3, lines 35-36), this is merely referring to the form of the if-else condition illustrated in columns 7-9, i.e. "case1: case 2: alternative 1;". There is nothing anywhere in Maslennikov that

even mentions predicate registers or loading values in predicate registers associated with branch addresses. Moreover, there is nothing in Maslennikov that teaches to call an instruction associated with one of the branch addresses based on values of a plurality of predicate registers. All Maslennikov is concerned with is optimizing switch instructions in code during compilation by moving the most probable switch alternatives out of the switch instruction and transforming them into predicate form before the switch instruction. Maslennikov has nothing to do with loading of predicate registers or calling instructions associated with branch addresses based on values of predicate registers. Thus, Maslennikov does not teach all of the features of independent claims 1, 11 and 20.

In view of the above, Applicant respectfully submits that Maslennikov does not teach each and every feature of independent claims 1, 11 and 20 as is required under 35 U.S.C. § 102(e). At least by virtue of their dependency on claims 1, 11 and 20, respectively, Maslennikov does not teach each and every feature of dependent claims 9, 18 and 28. Accordingly, Applicant respectfully requests withdrawal of the rejection of claims 1, 9, 11, 18, 20 and 28 under 35 U.S.C. § 102(e).

Furthermore, Maslennikov does not teach, suggest, or give any incentive to make the needed changes to reach the presently claimed invention. Absent the Examiner pointing out some teaching or incentive to implement Maslennikov to load a plurality of predicate registers with values associated with branch addresses based on one or more parameters and call an instruction associated with one of the branch addresses based on values of the plurality of predicate registers, one of ordinary skill in the art would not be led to modify Maslennikov to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion, or incentive to modify Maslennikov in this manner, the presently claimed invention can be reached only through an improper use of hindsight using the Applicant's disclosure as a template to make the necessary changes to reach the claimed invention.

VI. 35 U.S.C. § 103, Alleged Obviousness of Claims 7-8, 17 and 26-27

The Office Action rejects claims 7-8, 17 and 26-27 under 35 U.S.C. § 103(a) as being allegedly unpatentable over Maslennikov in view of The Java Virtual Machine

Specification (herein referred to as JVM Specification). This rejection is respectfully traversed for the same reasons as noted above with regard to the rejection under 35 U.S.C. § 102(e). That is, Maslennikov does not teach or suggest loading a plurality of predicate registers with values associated with branch addresses based on one or more parameters or calling an instruction associated with one of the branch addresses based on values of the plurality of predicate registers. Moreover, the JVM Specification does not teach or suggest these features either. Thus, since neither reference alone teaches or suggests these features, any alleged combination of these references would not result in these teachings or a suggestion to include such teachings.

In view of the above, Applicant respectfully submits that neither Maslennikov nor the JVM Specification, either alone or in combination, teach or suggest the features of independent claims 1, 11 and 20 from which claims 7-8, 17 and 26-27 depend. Accordingly, Applicant respectfully requests that the rejection of claims 7-8, 17 and 26-27 under 35 U.S.C. §103(a) be withdrawn.

VII. Conclusion

It is respectfully urged that the subject application is patentable over Maslennikov and JVM Specification and is now in condition for allowance. The Examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the Examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

Respectfully submitted,

DATE: December 19, 2003

Stephen J. Walder, Jr.
Reg. No. 41,534
Carstens, Yee & Cahoon, LLP
P.O. Box 802334
Dallas, TX 75380
(972) 367-2001
Attorney for Applicant